

Generalità sui sistemi microcontrollori

E programmazione scheda Arduino
UNO

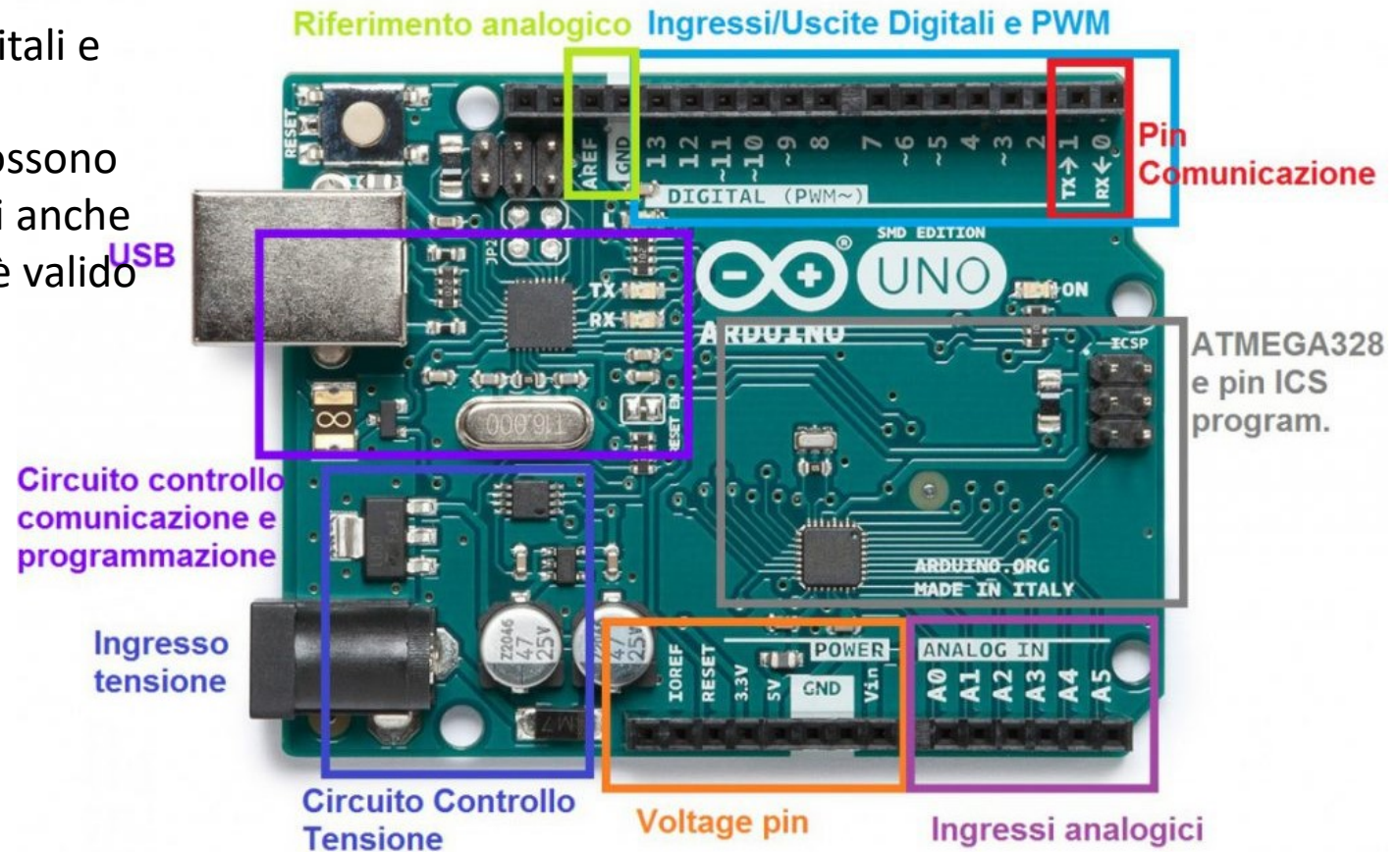
Microcontrollori e microprocessori

- Microprocessori: sistemi complessi dotati di CPU (Central Processing Unit); l'architettura è di tipo CISC (Complex Instruction Set Computing)
- Microcontrollori: sistemi semplici dotati di una semplice CPU (Central Processin Unit) una memoria interna; l'architettura è di tipo RISC (Reducted Instruction Set Computing)

La scheda di sviluppo Arduino UNO

- La scheda di sviluppo Arduino UNO è dotata di un microcontrollore (μC) Atmega 328 della Atmel

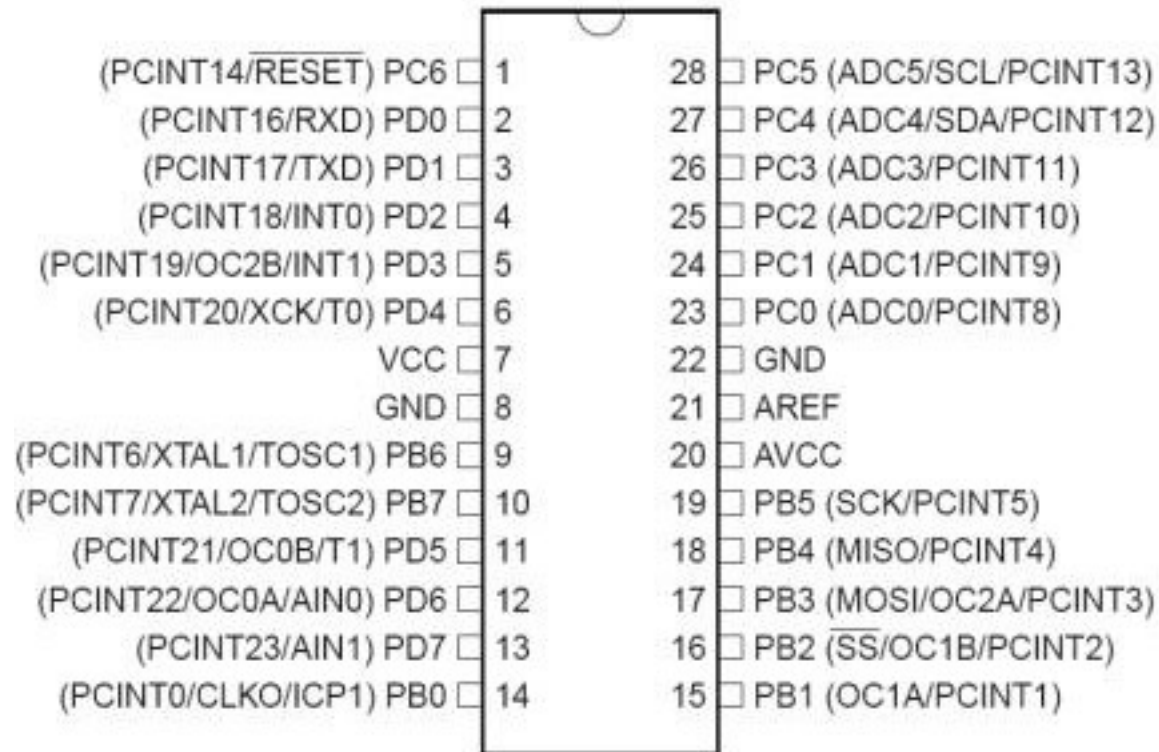
I pin sono 14 digitali e 6 analogici.
I pin analogici possono essere dichiarati anche digitali ma non è valido il viceversa



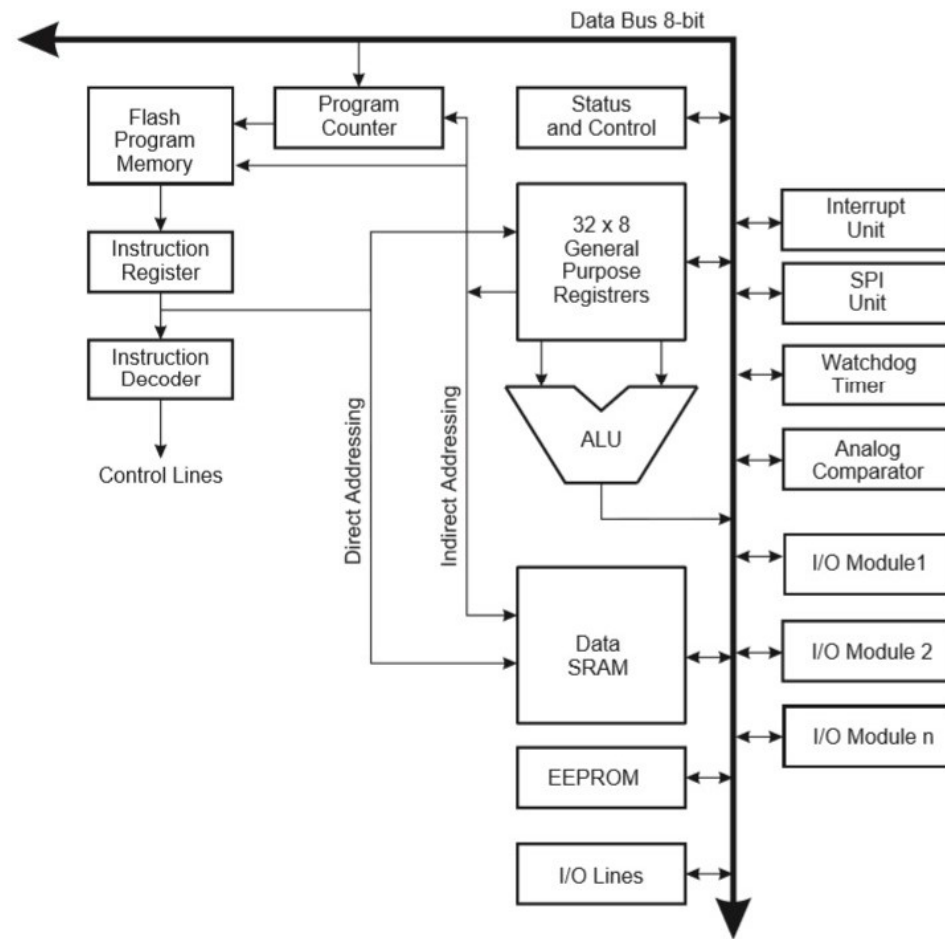
Specifiche hardware

Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

Microcontrollore atmega 328

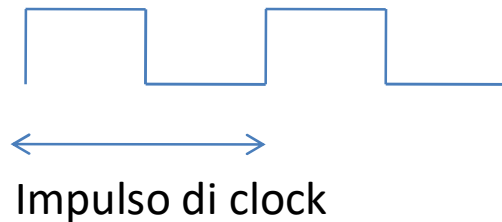


Architettura interna



Caricamento ed elaborazione dati

1. I dati vengono caricati in seriale tramite la porta USB(essendo caricati dall'IDE Arduino del computer);
2. Per ogni impulso di clock le istruzioni vengono decodificate e caricate nel registro istruzioni



Registri

3. I registri di uso generale sono a 8 bit ma ci sono anche 3 registri a 16 bit.

I registri a 8 bit vengono utilizzati per memorizzare i dati per calcoli e risultati normali. I registri a 16 bit vengono utilizzati per memorizzare i dati del contatore del timer in 2 registri diversi. Per esempio. X-basso e X-alto. Sono veloci e vengono utilizzati per memorizzare funzioni hardware specifiche.

4. La EEPROM memorizza i dati in modo permanente anche se l'alimentazione viene interrotta. La programmazione all'interno di una EEPROM è lenta

Registri particolari

5. L'Unità di interrupt controlla se c'è un interrupt per l'esecuzione dell'istruzione da eseguire in ISR (Interrupt Service Routine).
6. Serial Peripheral Interface (SPI) è un bus di interfaccia comunemente utilizzato per inviare dati tra microcontrollori e piccole periferiche come fotocamera, display, schede SD, ecc. Utilizza linee di clock e dati separate, insieme a una linea di selezione per scegliere il dispositivo con cui si desidera parlare.
7. Watchdog timer viene utilizzato per rilevare e ripristinare il malfunzionamento dell'MCU.
8. Il comparatore analogico confronta i valori di ingresso sul pin positivo e negativo, quando il valore del pin positivo è maggiore l'uscita viene impostata l'uscita a 1.
9. I registri di stato e di controllo vengono utilizzati per controllare il flusso di esecuzione dei comandi controllando ad intervalli regolari altri blocchi all'interno della CPU

ALU e pin I/O

10. ALU (Arithmetic and Logical unit) L'AVR ALU ad alte prestazioni opera in collegamento diretto con tutti i 32 registri di lavoro generici. All'interno di un singolo ciclo di clock, vengono eseguite le operazioni aritmetiche di una linea di registro di uso generale. Le operazioni ALU sono suddivise in 3 categorie principali: aritmetica, logica e funzioni digitali.
11. Pin I / O Gli ingressi e le uscite digitali (I / O digitali) su Arduino sono ciò che consente di collegare i sensori, gli attuatori e altri circuiti integrati di Arduino.

Extra

1. Reset: Ripristino automatico (software). Piuttosto che richiedere una pressione fisica del pulsante di ripristino prima di un caricamento, Arduino è progettato in modo da consentirne il ripristino tramite software in esecuzione su un computer collegato. Il software Arduino (IDE) utilizza questa funzionalità per consentire di caricare il codice semplicemente premendo il pulsante di caricamento nell'ambiente Arduino. Ciò significa che il bootloader può avere un timeout più breve, poiché l'abbassamento di DTR (Data Terminal Ready) può essere ben coordinato con l'avvio del caricamento.
2. Firmware: il firmware è un programma software o un insieme di istruzioni programmate su un dispositivo hardware. Fornisce le istruzioni necessarie su come il dispositivo comunica con l'altro hardware del computer. Il firmware è contenuto in dispositivi di memoria non volatile come la ROM.
3. To check se il firmware è installato nel tuo Arduino o meno basta premere il pulsante di reset e se il LED integrato lampeggia (sul pin 13) il firmware è presente.

Linguaggi di programmazione

- Per poter utilizzare un microcontrollore o un microprocessore bisogna utilizzare dei programmi
- Un programma è un file di testo che deve essere creato secondo delle regole ben precise attraverso un altro programma detto IDE (integrated development environment) che processa il file di testo per testare eventuali errori e, nel caso che tutto sia corretto, il file da testuale viene trasformato in ASCII

Linguaggi di programmazione

- Si dividono in linguaggi a basso livello e linguaggi ad alto livello
- Quelli a basso livello hanno una sintassi molto vicina a quella del microcontrollore o microprocessore
- Quelli ad alto livello sono molto vicini a linguaggio dell'uomo

Linguaggio di programmazione per la scheda Arduino

- È un linguaggio C/C++ dedicato per la scheda Arduino
- Non necessita di librerie basilari come per i linguaggi C/C++ tranne che per casi particolari
- È un linguaggio case sensitive nel senso che è sensibile alle lettere minuscole e alle lettere maiuscole
- Una volta compilato il programma, può essere trasferito sulla scheda tramite la porta USB (Universal Serial Bus)

IDE Arduino e template

sketch_sep30a | Arduino 1.8.13

File Modifica Sketch Strumenti Aiuto



sketch_sep30a

```
void setup() {  
  // put your setup code here, to run once:  
  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
  
}
```

Template

- Il template di un programma è lo scheletro del programma
- Il template è formato principalmente dai blocchi void `setup()` e `void loop()`
- Void sta ad indicare che i blocchi sono vuoti nel senso che non restituiscono valori
- Nel blocco void `setup` fondamentalmente, vengono dichiarati i pin di input e quelli di output, viene inizializzata la seriale; nel blocco void `loop` vengono immessi i comandi da eseguire all'infinito.
- Le variabili dichiarate all'esterno dei blocchi sono globali, vengono viste dai blocchi; quelle dichiarate all'interno di un blocco sono viste solo nel blocco ma non dagli altri e sono quindi locali.